BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents *will not* correct images, please do not report the images to the Image Problem Mailbox.

METHOD AND DEVICE FOR CALCULATING ELLIPTIC CURVE SCALAR MULTIPL

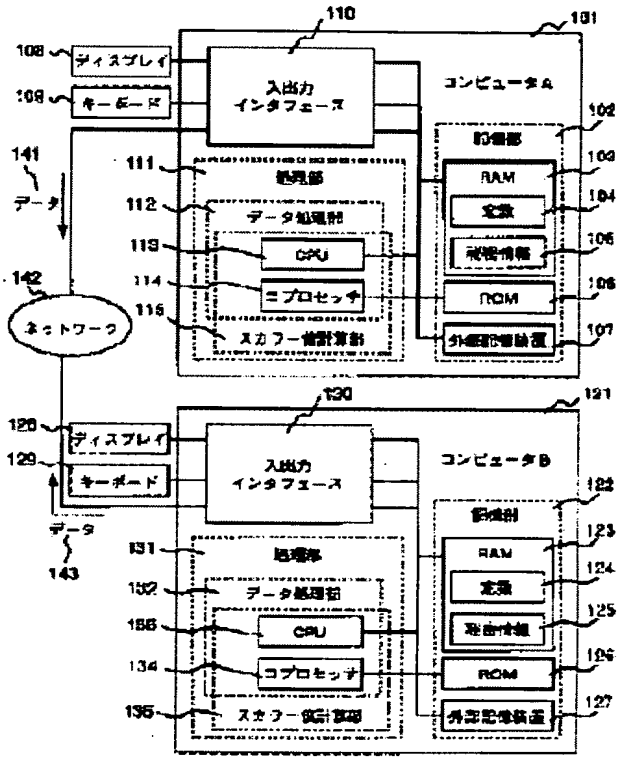
Patent number: JP2003255831
Publication date: 2003-09-10
Inventor: OKEYA KATSUYUKI; TANAKA HIDEYUKI
Applicant: HITACHI LTD
Classification:
- international: G09C1/00
- european:
Application number: JP20020052897 20020228
Priority number(s):

Abstract of **JP2003255831**

PROBLEM TO BE SOLVED: To provide an elliptic curve scalar multiple calculating method capable of preventing a side channel attack and a fault attack.

SOLUTION: In this scalar multiple calculating method for calculating a scalar multiple point from a scalar value and a point on an elliptic curve, a side channel attack is prevented by randomizing a given point and performing an elliptic curve calculation independent of a bit value in each bit of the scalar value. As for a fault attack, the fault attack is prevented by restoring the Y coordinate of the scalar multiple point and determining whether the scalar multiple point satisfies a definitional equation.

図 1



(19)日本国特許庁（ＪＰ）　(12)公開特許公報（Ａ）　(11)特許出願公開番号
特開2003－255831
（P2003－255831A）
(43)公開日　平成15年９月10日(2003.9.10)

(51)Int.Cl.⁷	識別記号	Ｆ Ｉ	テーマコード（参考）
Ｇ０９Ｃ　1/00	６５０	Ｇ０９Ｃ　1/00	６５０Ａ　５Ｊ１０４
	６２０		６２０Ａ

審査請求　未請求　請求項の数20　ＯＬ　（全 16 頁）

(21)出願番号　特願2002－52897(P2002－52897)

(22)出願日　平成14年２月28日(2002.2.28)

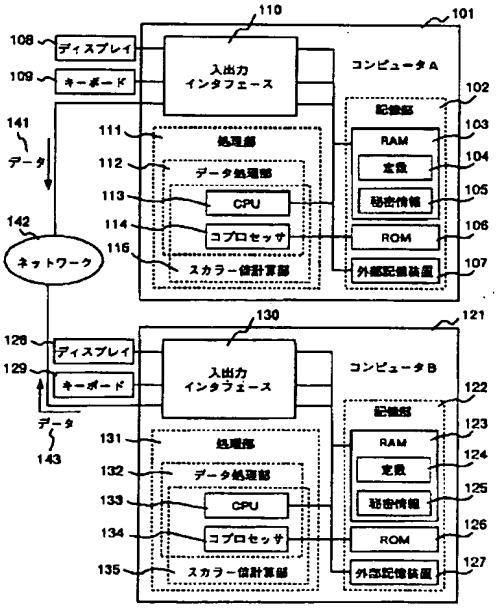
(71)出願人　000005108
株式会社日立製作所
東京都千代田区神田駿河台四丁目６番地
(72)発明者　桶屋　勝幸
神奈川県川崎市麻生区王禅寺1099番地　株式会社日立製作所システム開発研究所内
(72)発明者　田中　英幸
東京都小平市上水本町五丁目20番１号　株式会社日立製作所半導体グループ内
(74)代理人　100068504
弁理士　小川　勝男　（外２名）

Ｆターム(参考)　5J104 AA18 AA43 EA30 JA25 NA08 NA18

(54)【発明の名称】　楕円曲線スカラー倍計算方法及び装置

(57)【要約】
【課題】　サイドチャネル攻撃及びフォールト攻撃を防ぐことができる楕円曲線スカラー倍計算方法を提供すること。
【解決手段】　スカラー値及び楕円曲線上の点からスカラー倍点を計算するスカラー倍計算方法において、与えられた点をランダム化し、スカラー値のビットごとにビットの値とは独立の楕円曲線演算を行うことにより、サイドチャネル攻撃を防ぐ。またフォールト攻撃については、スカラー倍点のY座標を復元し、スカラー倍点が定義方程式をみたすか否かを判定することにより、フォールト攻撃をも防ぐ。

図　1



【特許請求の範囲】

【請求項１】楕円曲線暗号における楕円曲線についてスカラー値及び楕円曲線上の点からスカラー倍点を計算する方法であって、

前記楕円曲線上の点が所定の定義方程式を満たすか否かを判定する第１のステップと、

前記楕円曲線上の点をランダム化する第２のステップと、

前記スカラー値のビットの値を判定する第３のステップと、

前記スカラー値のビットごとにそのビット値によらない同一種別の演算を行う第４のステップと、

前記スカラー値のすべてのビットについて前記第３のステップ及び前記第４のステップを終了したとき、前記演算の結果に基づいて前記楕円曲線上の点の座標値を計算する第５のステップと、

計算された前記座標値が前記所定の定義方程式を満たすか否かを判定する第６のステップとを有することを特徴とするスカラー倍計算方法。

【請求項２】前記第４のステップは、前記ランダム化した点より導出された値と前記楕円曲線上の点をランダム化せずに導出された値との演算を実行するステップを含むことを特徴とする請求項１記載のスカラー倍計算方法。

【請求項３】前記スカラー値のすべてのビットについて前記第３のステップ及び前記第４のステップを実行する代わりに、前記スカラー値を格納し前記スカラー値のビット数より大きいビット数を備える記憶領域のすべてのビットについて前記第３のステップ及び前記第４のステップを実行することを特徴とする請求項１記載のスカラー倍計算方法。

【請求項４】前記楕円曲線としてモンゴメリ型楕円曲線を用いることを特徴とする請求項１記載のスカラー倍計算方法。

【請求項５】前記楕円曲線としてワイエルシュトラス型楕円曲線を用いることを特徴とする請求項１記載のスカラー倍計算方法。

【請求項６】前記楕円曲線として標数２の有限体上に定義された楕円曲線を用いることを特徴とする請求項１記載のスカラー倍計算方法。

【請求項７】前記楕円曲線としてＯＥＦ上に定義された楕円曲線を用いることを特徴とする請求項１記載のスカラー倍計算方法。

【請求項８】暗号化されたデータから復号データを生成する復号化方法であって、請求項１記載のスカラー倍計算方法を用いるステップを有することを特徴とする復号化方法。

【請求項９】署名データの正当性を検証する署名検証方法であって、請求項１記載のスカラー倍計算方法を用いるステップを有することを特徴とする署名検証方法。

【請求項１０】前記スカラー値は秘密鍵を示し、前記楕円曲線上の点は公開鍵を示すことを特徴とする請求項１記載のスカラー倍計算方法。

【請求項１１】楕円曲線暗号における楕円曲線についてスカラー値及び楕円曲線上の点からスカラー倍点を計算する装置であって、

前記楕円曲線上の点が所定の定義方程式を満たすか否かを判定する第１の処理手段と、

前記楕円曲線上の点をランダム化する第２の処理手段と、

前記スカラー値のビットの値を判定する第３の処理手段と、

前記スカラー値のビットごとにそのビット値によらない同一種別の演算を行う第４の処理手段と、

前記スカラー値のすべてのビットについて前記第３の処理手段及び前記第４の処理手段による処理を終了したとき、前記演算の結果に基づいて前記楕円曲線上の点の座標値を計算する第５の処理手段と、

計算された前記座標値が前記所定の定義方程式を満たすか否かを判定する第６の処理手段とを有することを特徴とするスカラー倍計算装置。

【請求項１２】前記第４の処理手段は、前記ランダム化した点より導出された値と前記楕円曲線上の点をランダム化せずに導出された値との演算を実行する処理手段を含むことを特徴とする請求項１１記載のスカラー倍計算装置。

【請求項１３】前記スカラー値のすべてのビットについて前記第３の処理手段及び前記第４の処理手段を実行する代わりに、前記スカラー値を格納し前記スカラー値のビット数より大きいビット数を備える記憶領域のすべてのビットについて前記第３の処理手段及び前記第４の処理手段を実行することを特徴とする請求項１１記載のスカラー倍計算装置。

【請求項１４】前記楕円曲線としてモンゴメリ型楕円曲線を用いることを特徴とする請求項１１記載のスカラー倍計算装置。

【請求項１５】前記楕円曲線としてワイエルシュトラス型楕円曲線を用いることを特徴とする請求項１１記載のスカラー倍計算装置。

【請求項１６】前記楕円曲線として標数２の有限体上に定義された楕円曲線を用いることを特徴とする請求項１１記載のスカラー倍計算装置。

【請求項１７】前記楕円曲線としてＯＥＦ上に定義された楕円曲線を用いることを特徴とする請求項１１記載のスカラー倍計算装置。

【請求項１８】暗号化されたデータから復号データを生成する復号化装置であって、請求項１１記載のスカラー倍計算装置を含むことを特徴とする復号化装置。

【請求項１９】請求項１１記載のスカラー倍計算装置を含むことを特徴とするＩＣカード。

【請求項２０】署名データの正当性を検証する署名検証装置であって、請求項１１記載のスカラー倍計算装置を含むことを特徴とする署名検証装置。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】本発明は、セキュリティ技術に係り、特に楕円曲線演算を用いたメッセージ処理方法に関する。

【０００２】

【従来の技術】楕円曲線暗号は、N.Koblitz, V.S.Millerにより提案された公開鍵暗号の一種である。公開鍵暗号には、公開鍵と呼ばれる一般に公開してよい情報と、秘密鍵と呼ばれる秘匿しなければならない秘密情報がある。与えられたメッセージの暗号化や署名の検証には公開鍵を用い、与えられたメッセージの復号化や署名の作成には秘密鍵を用いる。

【０００３】楕円曲線暗号における秘密鍵は、スカラー

$$x_3=B((y_2-y_1)/(x_2-x_1))^2-A-x_1-x_2$$

$$y_3=((y_2-y_1)/(x_2-x_1))(x_1-x_3)-y_1$$

を計算することにより得られる。ここでA, Bはモンゴメ

$$By^2=x^3+Ax^2+x$$

の係数である。

【０００６】楕円曲線上の点の２倍算は次のようにして計算される。楕円曲線上の点における接線をひくと、その接線は楕円曲線上の他の点において交わる。その交わった点とx座標に関して対称な点を、２倍算を行った結果の点とする。ある点に対し特定の回数だけ加法を行うことをスカラー倍といい、その結果をスカラー倍点、その回数のことをスカラー値という。

【０００７】楕円曲線上の離散対数問題の求解の困難性が理論的に確立されてきている一方で、実際の実装においては秘密鍵等の秘密情報に関連する情報（計算時間や電力消費量など）が暗号処理において漏洩する場合があり、その漏洩情報をもとに秘密情報を復元するといったサイドチャネル攻撃という攻撃法が提案されている。

【０００８】楕円曲線暗号に対するサイドチャネル攻撃が

文献１：J.Coron, Resistance against Differential Power Analysis for Elliptic Curve Cryptosystems, Cryptographic Hardware and Embedded Systems: Proceedings of CHES'99, LNCS 1717, Springer-Verlag, (1999) pp.292-302.

に記載されている。

【０００９】また暗号処理実行中にメモリ等に保持されている値が不正な値となることや、ＣＰＵでの計算中に計算間違いが発生すること、乃至は悪意ある攻撃者により故意にエラーを引き起こさせられることなどにより、エラーが発生した際の出力結果から秘密情報を推定するといったフォールト攻撃という攻撃法が提案されている。

値が担っている。また楕円曲線暗号の安全性は楕円曲線上の離散対数問題の求解が困難であることに由来している。楕円曲線上の離散対数問題とは、楕円曲線上のある点Pとそのスカラー倍の点dPが与えられた時、スカラー値dを求める問題である。

【０００４】楕円曲線上の点とは、楕円曲線の定義方程式をみたす数の組をいい、楕円曲線上の点全体には、無限遠点という仮想的な点を単位元とした演算、すなわち楕円曲線上の加法（乃至は加算）が定義される。そして同じ点同士による楕円曲線上の加法のことを、特に楕円曲線上の２倍算という。

【０００５】楕円曲線上の２点の加法は次のようにして計算される。２点を通る直線を引くとその直線は楕円曲線と他の点において交わる。その交わった点とx軸に関して対称な点を、加法を行った結果の点とする。例えばモンゴメリ型楕円曲線の場合には、点(x_1, y_1)と点(x_2, y_2)の加算$(x_3, y_3)=(x_1, y_1)+(x_2, y_2)$は、

（式１）

（式２）

リ型楕円曲線の定義式

（式３）

【００１０】楕円曲線暗号に対するフォールト攻撃が

文献２：I.Biehl, B.Meyer, V.Mueller, Differential Fault Attacks on Elliptic Curve Cryptosystems, Advances in Cryptology CRYPTO 2000, LNCS 1880, Springer-Verlag, (2000) pp.131-146.

に記載されている。

【００１１】楕円曲線暗号においては、与えられたメッセージの暗号化、復号化、署名の作成またはその検証は、楕円曲線演算を用いて行う必要がある。特に楕円曲線上のスカラー倍の計算は、秘密情報であるスカラー値を用いた暗号処理において用いられる。

【００１２】楕円曲線暗号に対するサイドチャネル攻撃の防御法が、

文献３：K.Okeya, K.Sakurai, Power Analysis Breaks Elliptic Curve Cryptosystems even Secure Against the Timing Attack, Progress in Cryptology - INDOCRYPT 2000, LNCS 1977, Springer-Verlag, (2000), pp.178-190.

に記載されている。

【００１３】モンゴメリ型楕円曲線を用いて行う楕円曲線上のスカラー倍計算において、与えられた楕円曲線上の点をランダム化することにより、サイドチャネル攻撃を防ぐという方法が提案されている。

【００１４】

【発明が解決しようとする課題】情報通信ネットワークの進展と共に電子情報に対する秘匿や認証のために暗号技術は不可欠な要素となってきている。スマートカード上で暗号技術を実装する場合、電力は外部より供給されるため攻撃者が電力消費量を観測することができるの

で、サイドチャネル攻撃を防ぐ必要性がある。しかしながら磁力や電力により攻撃者が故意にエラーを発生させることが可能なスマートカードや、大量の暗号処理を扱うため暗号処理中にエラーの発生する確率が高くなると想定されるサーバ等においては、フォールト攻撃についてもその攻撃にさらされる環境下にある。それゆえにサイドチャネル攻撃に加えフォールト攻撃をも防ぐ必要がある。

【００１５】上記技術は、サイドチャネル攻撃を防ぐ方法としては有効であるが、フォールト攻撃を防ぐという点は考慮されていない。

【００１６】本発明の目的は、サイドチャネル攻撃及びフォールト攻撃を防ぐことができる楕円曲線演算方法を提供することにある。

【００１７】本発明の他の目的は、上記楕円曲線演算方法を用いた暗号化処理方法、復号化処理方法、署名作成方法、署名検証方法を提供することにある。

【００１８】

【課題を解決するための手段】本発明は、楕円曲線演算において、スカラー値と楕円曲線上の点からスカラー倍点を計算するスカラー倍計算方法であって、楕円曲線上の点が所定の定義方程式を満たすか否かを判定する第１のステップと、楕円曲線上の点をランダム化する第２のステップと、スカラー値のビットの値を判定する第３のステップと、スカラー値のビットごとにそのビット値によらない同一種別の演算を行う第４のステップと、スカラー値のすべてのビットについて第３のステップ及び第４のステップを終了したとき、その演算の結果に基づいて楕円曲線上の点の座標値を計算する第５のステップと、計算された座標値が上記の所定の定義方程式を満たすか否かを判定する第６のステップとを有するスカラー倍計算方法を特徴とする。

【００１９】

【発明の実施の形態】以下、本発明の実施例について図面により説明する。

【００２０】図１は、ネットワーク142によって接続された本発明による楕円曲線演算方法を適用したコンピュータＡ101、コンピュータＢ121がネットワーク142により接続されたシステム構成を示すものである。

【００２１】図１の暗号通信システムにおけるコンピュータＡ101でメッセージの暗号化を行うには、$P_m+k(aQ)$及びkQを計算して出力し、コンピュータＢ121で暗号文の復号化を行うには、秘密鍵a及びkQより$-a(kQ)$を計算し、

$$(P_m+k(aQ))-a(kQ) \qquad (式４)$$

を計算して出力すればよい。ここでP_mはメッセージ、kは乱数（整数）、aは秘密鍵を示す定数（整数）、Qは定点、aQは公開鍵を示す点である。

【００２２】ネットワーク142には、$P_m+k(aQ)$, kQのみ送信され、メッセージP_mを復元するためには、kaQ、すなわちkQのa倍を計算する必要がある。ところが秘密鍵aはネットワーク142には送信されないため、秘密鍵aを保持しているものだけが、P_mを復元できることになる。

【００２３】図１において、コンピュータＡ101は、ＣＰＵ113やコプロセッサ114などの演算装置、ＲＡＭ103、ＲＯＭ106や外部記憶装置107などの記憶装置、コンピュータ外部とのデータ入出力を行う入出力インタフェース110を装備しており、外部にはコンピュータＡ101をユーザが操作するためのディスプレイ108、キーボード109、可搬型記憶媒体の読み書き装置などが接続されている。

【００２４】更にコンピュータＡ101は、ＲＡＭ103、ＲＯＭ106や外部記憶装置107などの記憶装置によって記憶部102を実現し、ＣＰＵ113やコプロセッサ114などの演算装置が記憶部102に格納されたプログラムを実行することによりデータ処理部112及びスカラー倍計算部115を実現する。データ処理部112は、本実施形態においては暗号化処理部として機能し、入力されたメッセージの暗号化を行う。スカラー倍計算部115は、データ処理部112で暗号化を行うのに必要なパラメタを計算する。記憶部102は、定数103（例えば楕円曲線の定義式や楕円曲線上の定点である）、秘密情報105（例えば秘密鍵である）などを記憶している。

【００２５】コンピュータＢ121は、コンピュータＡ101と同様のハードウェア構成を備える。更にコンピュータＢ121は、ＲＡＭ103、ＲＯＭ106や外部記憶装置107などの記憶装置によって記憶部122を実現し、ＣＰＵ113やコプロセッサ114などの演算装置が記憶部122に格納されたプログラムを実行することにより、データ処理部132及びスカラー倍計算部135を実現する。

【００２６】データ処理部132は、本実施形態においては復号化処理部として機能し、暗号化されたメッセージである暗号文141の復号化を行う。スカラー倍計算部135は、データ処理部132で復号化を行うのに必要なパラメータを計算する。記憶部122は、定数124（例えば楕円曲線の定義式や楕円曲線上の定点である）、秘密情報125（例えば秘密鍵である）などを記憶している。

【００２７】図２は、コンピュータＡ101の各処理部及び記憶部が行う情報の受け渡しの様子を示したものである。まずコンピュータＡ101が、入力されたメッセージを暗号化する場合の動作について説明する。メッセージはディジタル化されたデータであれば良く、テキスト、画像、映像、音などの種類は問わない。データ処理部112は、入出力インタフェース110を介して平文の入力メッセージ204を受け取ると、入力された平文メッセージのビット長が予め定めたビット長か否かを判断する。予め定めたビット長より長い場合には、予め定めたビット長となるように平文メッセージを区切る。以下、所定のビット長に区切られている部分メッセージ(単にメッセージともいう)について説明する。

【００２８】次にデータ処理部112は、メッセージのビット列によって表される数値をx座標(x_1)にもつ楕円曲線上の点P_aのy座標の値(y_1)を計算する。例えばモンゴメリ型楕円曲線は、B, Aはそれぞれ定数とするとき、

$$By_1^2=x_1^3+Ax_1^2+x_1 \qquad (式５)$$

で表されるので、これよりy座標の値を求めることができる。次にデータ処理部112は、乱数kを生成する。そして記憶部102に格納されている定数104から読み出した(図２の205)公開鍵aQとQのx座標と、求めたy座標の値と乱数kとをスカラー倍計算部115へ送る(図２の206)。

【００２９】スカラー倍計算部115は、Qのx座標、y座標の値、乱数kによるスカラー倍点$(x_{d1}, y_{d1})=kQ$と、公開鍵aQのx座標、y座標の値、乱数kによるスカラー倍点$(x$

$$x_{e1}=B((y_{d1}-y_1)/(x_{d1}-x_1))^2-A-x_1-x_{d1},$$

$$x_{e2}=x_{d2}$$

を計算し、暗号化されたメッセージx_{e1}, x_{e2}を得る。

【００３１】コンピュータＡ101は、データ処理部112で暗号化された1つ以上の部分メッセージから暗号化された出力メッセージ209を組み立てる。コンピュータＡ101は、暗号化された出力メッセージ209をデータ141として入出力インタフェース110より出力し、ネットワーク142を介してコンピュータＢ121へ転送する。

【００３２】なお図２の記憶部102からの情報読み出しは、スカラー計算部115へこの情報を送る前であれば入力メッセージを受け付ける前であっても良い。

【００３３】次にコンピュータＢ121が暗号化されたメッセージ141を復号化する場合の動作について、図２を参照しつつ説明する。ただしデータ処理部112をデータ処理部132、スカラー計算部115をスカラー倍計算部135、記憶部102を記憶部122と読み替えるものとする。また符号205では公開鍵の代わりに秘密鍵を読み出すものとする。

【００３４】データ処理部132は、入出力インタフェース110を介して暗号化されたデータ141(図２の入力メッセージ204) が入力されると、入力された暗号化されたデータ141のビット長が予め定めたビット長か否かを判断する。予め定めたビット長より長い場合には、予め定めたビット長となるように暗号化されたデータを区切

$$x_{f1}=B((y_{e1}+y_{d2})/(x_{e1}-x_{d2}))^2-A-x_{e1}-x_{d3}$$

を計算し、暗号化される前の部分メッセージx_1に相当するx_{f1}を得る。

【００３９】コンピュータＢ121は、データ処理部132で復号化された部分メッセージから平文メッセージを組み立て、入出力インタフェース110を介してディスプレイ108などへ出力する。

【００４０】次にコンピュータＢ121が復号化処理を行う場合のスカラー倍計算部135の処理を詳細に説明する。図３は、スカラー倍計算部135の機能ブロックを示したものである。スカラー倍計算部135は、ランダム化部302、加算部303、２倍算部304、ビット値判定部305、

$_{d2}, y_{d2})=k(aQ)$とを計算し、計算されたスカラー倍点をデータ処理部112へ送る(図２の208)。もし計算中にエラーが発生した場合は、「不正」を示す信号をデータ処理部112へ送る(図２の207)。データ処理部112は、スカラー倍計算部115より「不正」信号が送られてきた場合、必要であれば記憶部102に格納されている定数104を再び読み出し、再度公開鍵aQとQのx座標、y座標の値と乱数kとをスカラー倍計算部115へ送る(図２の206)。

【００３０】データ処理部112は、スカラー倍計算部115よりスカラー倍点が送られてきた場合、送られたスカラー倍点を用いて暗号化処理を行う。例えばモンゴメリ型の楕円曲線については、$P_a+k(aQ)$とkQを計算する。すなわち、

(式６)

(式７)

る。以下、所定のビット長に区切られている部分データ（単にデータともいう）について説明する。

【００３５】データ141のビット列によって表される数値をx座標にもつ楕円曲線上のy座標の値を計算する。暗号化されたメッセージがx_{e1}, x_{e2}のビット列であり、モンゴメリ型楕円曲線の場合、y座標の値(y_{e1})は

$$By_{e1}^2=x_{e1}^3+Ax_{e1}^2+x_{e1} \qquad (式８)$$

から得ることができる（ただし、B, Aはそれぞれ定数である）。

【００３６】データ処理部132は、記憶部122に格納されている秘密情報125から読み出した(図２の205)秘密鍵aと、x座標、y座標の値(x_{e1}, y_{e1})を、スカラー倍計算部135へ送る(図２の206)。

【００３７】スカラー倍計算部135は、x座標、y座標の値、秘密情報125からスカラー倍点$(x_{d3}, y_{d3})=a(x_{e2}, y_{e2})$を計算する。スカラー倍計算部135は、計算されたスカラー倍点をデータ処理部132へ送る(図２の207)。データ処理部132は、送られたスカラー倍点を用いて復号化処理を行う。

【００３８】例えば暗号化されたメッセージがx_{e1}, x_{e2}のビット列であり、モンゴメリ型の楕円曲線の場合は、

$$(P_a+k(aQ))-a(kQ)=(x_{e1}, y_{e1})-(x_{d3}, y_{d3})$$

を計算することにより達成する。すなわち、

(式９)

繰り返し判定部306、定義方程式判定部307およびY座標復元部308からなる。

【００４１】図４及び図５を用いてスカラー倍計算部135がスカラー値d及びモンゴメリ型楕円曲線上の点Pからモンゴメリ型楕円曲線におけるスカラー倍点dPを計算する方法(第1の計算方法という)を説明する。スカラー倍計算部135がデータ処理部132からスカラー値dと楕円曲線上の点Pを受け取ると、定義方程式判定部307は、入力された点Pが楕円曲線上にあるか否かを判定する。これは入力された点Pが楕円曲線の定義式(式３)を満たすかどうかで判定する。満たす場合はステップ402へ、満た

さない場合はステップ524へ行く（401）。

【００４２】ステップ401で満たす場合、ランダム化部302は、入力された楕円曲線上の点Pのランダム化を行う。これはランダム化部302が次のような処理を行うことにより達成される。乱数rを生成し（402）、点P=(x, y)を射影座標においてランダム化された点をP=(rx, ry,

$$4X_1Z_1=(X_1+Z_1)^2-(X_1-Z_1)^2$$

$$X_2=(X_1+Z_1)^2(X_1-Z_1)^2$$

$$Z_2=(4X_1Z_1)((X_1-Z_1)^2+((A+2)/4)(4X_1Z_1))$$

であり、Aは定数、X_1, Z_1, X_2, Z_2はそれぞれ点PのX座標、Z座標、点2PのX座標、Z座標である。ランダム化された点Pとステップ405で求めた点2Pからなる点の組(P, 2P)をm=1の点の組(mP, (m+1)P)として(mは自然数)、記憶部122に一時的に格納する(406)。

【００４４】繰り返し判定部306は、変数Iと記憶部122から読み出したスカラー値dのビット長とが一致するかどうかを判定する(411)。一致すればステップ521へ行く。一致しなければステップ412へ行く。ステップ411で

$$X_{2m+1}=[(X_m-Z_m)(X_{m+1}+Z_{m+1})+(X_m+Z_m)(X_{m+1}-Z_{m+1})]^2,$$

$$Z_{2m+1}=x[(X_m-Z_m)(X_{m+1}+Z_{m+1})-(X_m+Z_m)(X_{m+1}-Z_{m+1})]^2$$

を計算することにより達成される。ここでX_m, Z_m, X_{m+1}, Z_{m+1}, X_{2m+1}, Z_{2m+1}はそれぞれ点mPのX座標、Z座標、点(m+1)PのX座標、Z座標、点(2m+1)PのX座標、Z座標である。

$$4X_mZ_m=(X_m+Z_m)^2-(X_m-Z_m)^2$$

$$X_{2m}=(X_m+Z_m)^2(X_m-Z_m)^2$$

$$Z_{2m}=(4X_mZ_m)((X_m-Z_m)^2+((A+2)/4)(4X_mZ_m))$$

を計算することにより達成される。ここでAは定数、X_m, Z_m, X_{2m}, Z_{2m}はそれぞれ点mPのX座標、Z座標、点2mPのX座標、Z座標である。ステップ415で求めた点2mPとステップ414で求めた点(2m+1)Pからなる点の組(2mP, (2m+1)P)を点の組(mP, (m+1)P)と置き換え、mに2mを代入し、ステップ411へ戻る（416）。

$$X_{2m+1}=[(X_m-Z_m)(X_{m+1}+Z_{m+1})+(X_m+Z_m)(X_{m+1}-Z_{m+1})]^2,$$

$$Z_{2m+1}=x[(X_m-Z_m)(X_{m+1}+Z_{m+1})-(X_m+Z_m)(X_{m+1}-Z_{m+1})]^2$$

を計算することにより達成される。

【００４９】２倍算部304は、射影座標で表された点の

$$4X_{m+1}Z_{m+1}=(X_{m+1}+Z_{m+1})^2-(X_{m+1}-Z_{m+1})^2$$

$$X_{2m+2}=(X_{m+1}+Z_{m+1})^2(X_{m+1}-Z_{m+1})^2$$

$$Z_{2m+2}=(4X_{m+1}Z_{m+1})((X_{m+1}-Z_{m+1})^2+((A+2)/4)(4X_{m+1}Z_{m+1}))$$

を計算することにより達成される。ここでAは定数、X_{m+1}, Z_{m+1}, X_{2m+2}, Z_{2m+2}はそれぞれ点(m+1)PのX座標、Z座標、点(2m+2)PのX座標、Z座標である。ステップ417で求めた点(2m+1)Pとステップ418で求めた点(2m+2)Pからなる点の組((2m+1)P, (2m+2)P)を、点の組(mP, (m+1)P)と置き換え、mに2m+1を代入し、ステップ411へ戻る（419）。ここでステップ417～419は、通常のアルゴリズムとは異なり、ステップ414～416と同一種別の演算を行っている。

【００５０】ステップ411で変数Iとスカラー値dのビット長とが一致した場合、Y座標復元部308は、点mPのY座標Y_mを復元する(521)。Y座標復元方法については、

文献４：K. Okeya, K. Sakurai, Efficient Elliptic Curve Cryptosystems from a Scalar Multiplication Algorithm with Recovery of the y-Coordinate ona Montgomery-Form Elliptic Curve, Cryptographic Hardware and Embedded Systems: Proceedings of CHES 2001, LNCS 2162, Springer-Verlag, (2001) pp. 126-141.

に記載されている。

【００５１】定義方程式判定部307は、点mPが楕円曲線上にあるかを判定する。これは点mPが楕円曲線の定義式(式３)を満たすかどうかで判定する。満たす場合はステ

r)と表す（403）。次に変数Iに初期値1を代入する（404）。

【００４３】２倍算部304は、ランダム化された点Pの２倍点2Pを、モンゴメリ型楕円曲線の射影座標における２倍算の公式を用いて計算する（405）。モンゴメリ型楕円曲線の射影座標における２倍算の公式は、

（式１０）

（式１１）

（式１２）

一致しなかった場合は、変数Iを1増加させる（412）。ビット値判定部305は、スカラー値dのI番目のビットの値が0であるか1であるかを判定し、0であればステップ414へ、1であればステップ417へ行く（413）。

【００４５】ステップ413でビットの値が0であった場合、加算部303は、射影座標により表された点の組(mP, (m+1)P)から点mPと点(m+1)Pの加算mP+(m+1)Pを、ランダム化されていない点P=(x; y)を利用して行い、点(2m+1)Pを計算する（414）。これは、

（式１３）

（式１４）

【００４６】２倍算部304は、射影座標で表された点の組(mP, (m+1)P)から点mPの2倍算2(mP)を行い、点2mPを計算する（415）。これは、

（式１５）

（式１６）

（式１７）

【００４７】ステップ413でビットの値が1であった場合、加算部303は、射影座標により表された点の組(mP, (m+1)P)から点mPと点(m+1)Pの加算mP+(m+1)Pをランダム化されていない点P=(x, y)を利用して行い、点(2m+1)Pを計算する（417）。

【００４８】これは、

（式１８）

（式１９）

組(mP, (m+1)P)から点(m+1)Pの２倍算2((m+1)P)を行い、点(2m+2)Pを計算する（418）。これは、

（式２０）

（式２１）

（式２２）

ップ523へ、満たさない場合はステップ524へ行く(522)。ステップ522で満たす場合、射影座標で表された点の組(mP, (m+1)P)から射影座標で表された点mP＝(X_m, Y_m, Z_m)をスカラー倍点dPとして復号化処理部132へ出力する(523)。

【００５２】ステップ401で満たさない場合、もしくはステップ522でみたさない場合、「不正」を示す信号を復号化処理部132へ出力する(524)。

【００５３】ここでスカラー倍点dPをアフィン座標等へ座標を変換して出力してもよい。またワイエルシュトラス型楕円曲線における座標に変換して出力してもよい。

【００５４】また上記手順により、mとスカラー値dはビット長が等しくさらにそのビットのパターンも同じとなるため等しくなる。そのため上記手順によりスカラー倍dPが計算できていることになる。

【００５５】なおスカラー倍計算部135に入力される楕円曲線上の点をモンゴメリ型楕円曲線上の点としたが、ワイエルシュトラス型楕円曲線上の点であってもよい。この場合は、ワイエルシュトラス型楕円曲線上の点をモンゴメリ型楕円曲線上の点に変換して用いればよい。

【００５６】上記方法は、サイドチャネル攻撃に対する防御に関しても有効である。この理由は次の通りである。ステップ403においてランダム化した点Pをそれ以降のステップで用いている。ステップ414及びステップ417はランダム化されていない点Pを用いるが、ステップ414及びステップ417では、ランダム化された点Pから導出された点mP及び(m+1)Pと、ランダム化されていない点Pとを用いて、演算を行い点(2m+1)Pを計算する。ステップ402の乱数生成で別の値が生成され、ステップ403でランダム化された点Pの座標の値が異なれば、ステップ414及びステップ417での点mP及び点(m+1)Pの座標の値が異なり、それらの値を用いて計算される点(2m+1)Pの座標の値も異なる。すなわち、同じスカラー値d及び点Pを与えても、その都度点(2m+1)Pの座標の値が変化する。

【００５７】さらにステップ413でのビットの値の判定結果にかかわらず同一の計算手順を踏むため、計算の実行順序とビットの値との間に依存関係がないことが分かる。

【００５８】この計算方法を実装する際には、ステップ413以降の処理について同じプログラムまたは処理回路をビット値に係わらず共有するように作成しても構わない。

【００５９】以上の通り、上記第1の計算方法は、サイドチャネル攻撃に有用な情報を与えないので、サイドチャネル攻撃に対して耐性がある。また用いている楕円曲線の性質により、高速に計算できるという特徴がある。

【００６０】また上記方法は、フォールト攻撃に対する防御に関しても有効である。この理由は次の通りである。まずスカラー倍計算部202に入力される楕円曲線上の点Pとして不正な値が与えられたとすると、ステップ401において点Pが楕円曲線の定義式(式３)をみたさないと判定され、その結果ステップ524で「不正」を出力する。次に計算途中に楕円曲線上の任意の点の値が不正な値となった場合、ステップ524で「不正」を出力する。このことを示す。

【００６１】ステップ402-419で点mPもしくは(m+1)Pの値が不正な値になったとする。その場合、次にくるステップ411で点mPもしくは点(m+1)Pの値が不正となる。ステップ411で点mPもしくは点(m+1)Pの値が不正であり、Iとスカラー値のビット長が等しくないと判定される場合、その次にくるステップ411でも点mPもしくは点(m+1)Pの値も不正となる。ステップ411で点mPの値が不正であり、Iとスカラー値のビット長が等しいと判定される場合、ステップ522で点mPは定義方程式(式３)を満たさないと判定され、ステップ524で「不正」を出力する。今度はステップ411で点mPの値が正しく、点(m+1)Pの値が不正であり、Iとスカラー値のビット長が等しいと判定される場合について考察する。ステップ521で点mPのY座標復元を行う。その際点(m+1)Pの値を用いるので、復元されたY座標の値は不正な値となる。したがってステップ522で点mPは定義方程式(式３)を満たさないと判定され、ステップ524で「不正」を出力する。

【００６２】最後に点Pとして正しい値が入力され、計算途中で不正な値となることが起こらなかった場合、すなわち常に正しい値であった場合、ステップ522で点mPは定義方程式(式３)を満たす。

【００６３】以上の通り、上記第1の計算方法は、フォールト攻撃に有用な情報を与えないので、フォールト攻撃に対して耐性がある。

【００６４】なお第1の計算方法では楕円曲線として、モンゴメリ型楕円曲線を用いたが、標数２の有限体上定義された楕円曲線を用いてもよいし、ＯＥＦ(Optimal Extension Field)上で定義された楕円曲線を用いてもよい。ＯＥＦについては、

文献５：D. V. Bailey, C. Paar, Optimal Extension Fields for Fast Arithmeticin Public-Key Algorithms, Advances in Cryptology CRYPTO '98, LNCS 1462, Springer-Verlag, (1998), pp. 472-485.

に記載されている。

【００６５】以上、コンピュータＢ121が暗号化されたデータ141を復号化する場合のスカラー倍計算部135の動作を説明したが、コンピュータＡ101が入力メッセージを暗号化する場合も同様である。

【００６６】その場合には、コンピュータＡ101のスカラー倍計算部115は、既に説明した楕円曲線上の点Q、乱数kによるスカラー倍点kQと、公開鍵aQと乱数kによるスカラー倍点k(aQ)を出力する。このとき第１の計算方法で説明したスカラー値dを乱数k、楕円曲線上の点Pを楕円曲線上の点Q、公開鍵aQとして同様の処理を行うことにより、それぞれのスカラー倍点を求めることができ

る。

【００６７】次に図６を用いて、スカラー倍計算部135が実ビット長Lのスカラー値d及びモンゴメリ型楕円曲線上の点Pから、モンゴメリ型楕円曲線におけるスカラー倍点dPを計算する方法(第２の計算方法という)を説明する。ここで実ビット長とは、スカラー値dの格納されている領域(メモリ、レジスタなど)のビット数であり、実ビット長≧スカラー値のビット長である。そのため最上位ビットは1でなくてもよい。

【００６８】この方法では、計算ステップと掛かる時間とがスカラー値dにかかわらず一定になるように構成している。これにより上記攻撃法に対して有用な情報を与えず、耐性を備えるようにしている。

【００６９】スカラー倍計算部135がデータ処理部132から楕円曲線上の点Pとスカラー値dを受け取ると、スカラー値dが0であるかどうかを判定し、0であれば無限遠点を出力し終了する。スカラー値dが0でなければ、処理を続ける（601）。定義方程式判定部307は、入力された点Pが楕円曲線上にあるかを判定する。これは入力された点Pが楕円曲線の定義式(式３)を満たすかどうかで判定する。満たす場合はステップ603へ、満たさない場合はステップ617へ行く（602）。ランダム化部302は、入力された楕円曲線上の点Pのランダム化を行う。すなわち乱数rを生成し（603）、ランダム化した点Pを射影座標において、(rx, ry, r)と表す（604）。

【００７０】次に楕円曲線上の不定点$T_{0,0}$, $T_{0,1}$, $T_{1,0}$, $T_{1,1}$の初期化を行う。$T_{0,0}$にはステップ604でランダム化された点Pを、$T_{1,0}$には$T_{0,0}$を、$T_{0,1}$にはステップ604でランダム化された点Pの２倍点2Pを、$T_{1,1}$には$T_{0,1}$を、それぞれ代入する。ランダム化された点Pの２倍点2Pの計算には、モンゴメリ型楕円曲線の射影座標における２倍算の公式（式１０、式１１、式１２）を用いて計算する（605）。

【００７１】次に変数sに初期値0を代入する（606）。変数iに初期値L-1を代入する（607）。繰り返し判定部306は、変数iが0以上であるかどうかを判定する。0以上のときはステップ609へ、0未満のときはステップ614へ行く（608）。楕円曲線上の不定点Tに点$T_{s,di}$を代入する(609)。d_iは、スカラー値dを$d=\Sigma d_j 2^j$, $d_j \in d\{0, 1\}$, jは0からL-1を動く、と表した時の、j=iの時のビットd_iである。２倍算部304は、射影座標で表された点Tの２倍算2(T)を行い、点2Tを点$T_{s,di}$に格納する（610）。加算部303は射影座標により表された点Tと射影座標により表された点$T_{s,1-di}$との加算をランダム化されていない点T=(x, y)を用いて行い、その結果を点$T_{s,1-di}$に格納する（611）。次にsとd_iとの論理和をとり、その結果をsに格納する（612）。次に変数iを1減少させる（613）。

【００７２】ステップ608でi<0の場合、Y座標復元部308は、点$T_{0,0}$及び点$T_{1,0}$のY座標を復元する（614）。Y座標復元方法については、文献４に記載されている。定義方程式判定部307は、点$T_{0,0}$及び点$T_{1,0}$が楕円曲線上にあるかを判定する。これは点$T_{0,0}$及び点$T_{1,0}$が楕円曲線上の定義式(式３)を満たすかどうかで判定する。ともに満たす場合はステップ616へ、いずれか一方が満たさない場合はステップ617へ行く（615）。ステップ615でともに満たす場合、射影座標で表された点$T_{1,0}$をスカラー倍点dPとして復号化処理部132へ出力する（616）。ここでアフィン座標等へ座標を変換して出力してもよい。またワイエルシュトラス型楕円曲線における座標に変換して出力してもよい。ステップ602で満たさない場合、もしくはステップ615でいずれか一方が満たさない場合、「不正」を示す信号をデータ処理部132へ出力する（617）。

【００７３】なおスカラー倍計算部202に入力される楕円曲線上の点をモンゴメリ型楕円曲線上の点としたが、ワイエルシュトラス型楕円曲線上の点であってもよい。この場合は、ワイエルシュトラス型楕円曲線上の点をモンゴメリ型楕円曲線上の点に変換して用いればよい。

【００７４】上記第２の計算方法も、サイドチャネル攻撃に対する防御に関して有効である。この理由は次の通りである。ステップ604においてランダム化した点Pをそれ以降のステップで用いている。ステップ611はランダム化されていない点Pを用いるが、ステップ611では、ランダム化された点Pから導出された点T及び点$T_{s,1-di}$とランダム化されてない点Pを用いてT+$T_{s,1-di}$を計算する。ステップ603の乱数生成で別の値が生成され、ステップ604でランダム化された点Pの座標の値が異なれば、ステップ611でのT及び$T_{s,1-di}$の座標の値が異なり、それらの値を用いて計算されるT+$T_{s,1-di}$の座標の値も異なる。すなわち同じスカラー値d及び点Pを与えても、そのつどT+$T_{s,1-di}$の座標の値が変化する。さらに各ビットd_iの値にかかわらず、同一の計算手順を踏むため、計算の実行順序とビットの値との間に依存関係がない。そのうえステップ608～ステップ613の繰り返し回数はdのビット長に依存せずに、必ずL回となるため、dのビット長にも依存しない。

【００７５】なおステップ606において、sにd_{L-1}を代入し、ステップ607において、IにL-2を代入してもよい。これによりスカラー値dの最上位ビットd_{L-1}が1である場合にダミー演算が発生しない。すなわち、s=0, i=L-1 の時に行っていたステップ608～613の最初の繰り返しを行わなくてもよくなり、さらなる高速化が可能になる。

【００７６】以上の通り、上記の第２の計算方法は、サイドチャネル攻撃に有用な情報を与えないので、サイドチャネル攻撃に対して耐性がある。

【００７７】また上記第２の計算方法も、フォールト攻撃に対する防御に関しても有効である。この理由は次の通りである。まずスカラー倍計算部202に入力される楕円曲線上の点Pとして不正な値が与えられたとすると、

ステップ602において点Pが楕円曲線の定義式(式３)を満たさないと判定され、その結果ステップ617で「不正」を出力する。

【００７８】次に計算途中に楕円曲線上の任意の点の値が不正な値となった場合、ステップ617で「不正」を出力する。このことを示す。ステップ603-613で点$T_{0,0}$, $T_{0,1}$, $T_{1,0}$, $T_{1,1}$のいずれかの値が不正な値になったとする。その場合、次にくるステップ608で点$T_{0,0}$, $T_{0,1}$, $T_{1,0}$, $T_{1,1}$のいずれかの値が不正となる。ステップ608で点$T_{0,0}$, $T_{0,1}$, $T_{1,0}$, $T_{1,1}$のいずれかの値が不正であり、Iとスカラー値のビット長が等しくないと判定される場合、その次にくるステップ608でも点$T_{0,0}$, $T_{0,1}$, $T_{1,0}$, $T_{1,1}$のいずれかの値も不正となる。ステップ608で点$T_{0,0}$乃至は点$T_{1,0}$の値が不正であり、Iとスカラー値のビット長が等しいと判定される場合、ステップ615で点$T_{0,0}$乃至は点$T_{1,0}$は定義方程式(式３)を満たさないと判定され、ステップ617で「不正」を出力する。今度はステップ608で点$T_{0,0}$, $T_{1,0}$, $T_{1,1}$の値が正しく、点$T_{0,1}$の値が不正であり、Iとスカラー値のビット長が等しいと判定される場合について考察する。ステップ614で点$T_{0,0}$のY座標復元を行う。その際点$T_{0,1}$の値を用いるので、復元されたY座標の値は不正な値となる。したがってステップ615で点$T_{0,0}$は定義方程式(式３)をみたさないと判定され、ステップ617で「不正」を出力する。ステップ608で点$T_{0,0}$, $T_{0,1}$, $T_{1,0}$の値が正しく、点$T_{1,1}$の値が不正であり、Iとスカラー値のビット長が等しいと判定される場合についても同様である。

【００７９】最後に点Pとして正しい値が入力され、計算途中で不正な値となることが起こらなかった場合、すなわち常に正しい値であった場合、ステップ615で点$T_{0,0}$及び点$T_{1,0}$は定義方程式(式３)を満たす。

【００８０】以上の通り、上記第２の計算方法は、フォールト攻撃に有用な情報を与えないので、フォールト攻撃に対して耐性がある。

【００８１】なお第２の計算方法では楕円曲線として、モンゴメリ型楕円曲線を用いたが、標数２の有限体上定義された楕円曲線を用いてもよいし、ＯＥＦ(Optimal Extension Field)上定義された楕円曲線を用いてもよい。ＯＥＦについては、文献５に記載されている。

【００８２】次に本発明を署名検証システムに適用する例について図７と図２を用いて説明する。図７の署名検証システムは、スマートカード701と署名検証処理を行うコンピュータ721とから成る。

【００８３】スマートカード701は、機能としてはコンピュータＡ101と類似の構成を備え、ＣＰＵ713やコプロセッサ714などの演算装置を備えているが、記憶部702に格納されているプログラムによってデータ処理部112ではなく署名生成処理部712を実現する。また外部記憶装置、ディスプレイおよびキーボードを備えない。

【００８４】コンピュータ721は、コンピュータＢ121と同様の構成を備えるが、記憶部722に格納されるプログラムによってデータ処理部112ではなく署名検証処理部732を実現する。

【００８５】スカラー倍計算部715と735は、各々図１に示すスカラー倍計算部115または135と同様の機能を備える。図６の署名検証システムにおける署名作成と署名検証動作について図２を参照して説明する。ただしデータ処理部112を署名生成処理部712、スカラー計算部115をスカラー倍計算部715、記憶部102を記憶部702と読み替えるものとする。また符号205では公開鍵はない。またコンピュータ721にあっては、データ処理部112を署名検証処理部732、スカラー計算部115をスカラー倍計算部735、記憶部102を記憶部722と読み替えるものとする。

【００８６】コンピュータ721は、ランダムに選んだ数値をチャレンジコード743として、インタフェース742を介してスマートカード701に転送する。署名生成処理部712は、チャレンジコード743を入力メッセージ204として受け付け、チャレンジコード743のハッシュ値をとり、所定のビット長の数値fに変換する。次に署名生成処理部712は、乱数uを生成し、記憶部702(図２の203)に格納されている定数704から読み出した(図２の205)楕円曲線上の定点Qとともにスカラー倍計算部715(図２の202)へ送る(図２の206)。

【００８７】スカラー倍計算部715は、定点Q、乱数uによるスカラー倍点(x_u, y_u)を計算し、計算されたスカラー倍点を署名生成処理部712へ送る(図２の208)。署名生成処理部712は、送られたスカラー倍点を用いて署名の生成を行う。例えばＥＣＤＳＡ署名であれば、

$$s = x_u \bmod q \qquad \text{(式２３)}$$

$$t = u^{-1}(f+ds) \bmod q \qquad \text{(式２４)}$$

を計算することによりチャレンジコード743に対応する署名(s, t)を得る。

【００８８】ここでqは定点Qの位数、すなわち定点Qのq倍点qQが無限遠点になり、qより小さな数値mに対する定点Qのm倍点mQは無限遠点にはならない、というような数値のことである。またu^{-1}は法qにおける逆数、すなわち$uu^{-1} = 1 \bmod q$となる数である。またdは秘密鍵を示す定数である。

【００８９】ＥＣＤＳＡ署名については、

文献６：ANSI X9.62 Public Key Cryptography for the Financial Services Industry, The Elliptic Curve Digital Signature Algorithm (ECDSA), (1998)

に記載されている。

【００９０】スマートカード701は、署名生成処理部712で作成した署名741を入出力インタフェース710より出力メッセージ209として出力し、インタフェース742を介してコンピュータ721へ転送する。コンピュータ721の署名検証処理部732は、署名741が入力される(図２の204)と、署名741の数値s, tが適切な範囲内すなわち$1 \leqq s$, $t < q$であるかを調べる。数値s, tが上記範囲内になければ

チャレンジコード743に対する署名の検証結果として「無効」を出力し、スマートカード701を拒絶する。数値s，tが上記範囲内にあれば、署名検証処理部732は、

$h = t^{-1} \bmod q$ （式２５）

$h_1 = fh \bmod q$ （式２６）

$h_2 = sh \bmod q$ （式２７）

を計算する。そして記憶部722に格納されている定数724から読み出した(図２の205)公開鍵aQ及び定点Qと計算したh_1, h_2をスカラー倍計算部735へ送る(図２の206)。

【００９１】スカラー倍計算部735は、定点Qとh_1によるスカラー倍点h_1Qと、公開鍵aQとh_2によるスカラー倍点h_2aQとを計算し、計算されたスカラー倍点を署名検証処理部732へ送る(図２の208)。

【００９２】署名検証処理部732は、送られたスカラー倍点を用いて署名検証処理を行う。例えば点R

$R=h_1Q+h_2aQ$ （式２８）

を計算し、そのx座標をx_Rとしたとき、

$s' = x_R \bmod q$ （式２９）

を計算し、s'=sであればチャレンジコード743に対する署名の検証結果として「有効」を出力し、スマートカード701を認証し、受け入れる。s'=sでなければ「無効」を出力し、スマートカードを拒絶する。

【００９３】上記実施形態のスカラー倍計算部715、735は、図１のスカラー倍計算部115または135と同様の機能を備えるので、サイドチャネル攻撃及びフォールト攻撃を防ぐスカラー倍計算を実行できる。そのためスマートカード701は署名作成処理を行う際に、コンピュータ721は署名検証処理を行う際に、サイドチャネル攻撃及びフォールト攻撃を防いだ実行ができる。

【００９４】次に本発明を鍵交換システムに適用する例を説明する。本実施形態においては、図１のシステム構成が応用できる。図１のデータ処理部112、132は、本実施形態においては、それぞれ鍵交換処理部として機能する。鍵交換システムのコンピュータＡ101が入力されたデータ143から共有情報の導出を行う場合の動作について図１、図２を参照して説明する。ただしコンピュータＢ121にあってはデータ処理部112をデータ処理部132、スカラー計算部115をスカラー倍計算部135、記憶部102を記憶部122と読み替えるものとする。また両コンピュータＡ101、コンピュータＢ121において符号205では公開鍵の代わりに秘密鍵を読み出すものとする。

【００９５】コンピュータＢ121のデータ処理部132は、記憶部122の秘密情報125から秘密鍵bを読み出しコンピュータＢ121の公開鍵bQを計算する。そしてネットワーク142を介して公開鍵bQをデータ143としてコンピュータＡ101に転送する。

【００９６】コンピュータＡ101のデータ処理部112は、コンピュータＢ121の公開鍵bQの入力を入力メッセージ204として受け付けると、データ処理部112は、記憶部102から読み出した(図２の205)秘密情報105であるコンピュータＡ101の秘密鍵aと、コンピュータＢ121の公開鍵bQとをスカラー倍計算部115へ送る(図２の206)。

【００９７】スカラー倍計算部115は、秘密鍵aと公開鍵bQによるスカラー倍点abQを計算し、計算されたスカラー倍点をデータ処理部112へ送る(図２の208)。データ処理部112は、送られたスカラー倍点を用いて共有情報の導出を行い、記憶部102に秘密情報105として格納する。例えばスカラー倍点abQのx座標を、共有情報とする。

【００９８】次にコンピュータ121が、入力されたデータ141から共有情報の導出を行う場合の動作について説明する。コンピュータＡ101のデータ処理部112は、記憶部102の秘密情報105から秘密鍵aを読み出しコンピュータＡ101の公開鍵aQを計算する。そしてネットワーク142を介して公開鍵aQをデータ141としてコンピュータＢ121に転送する。

【００９９】コンピュータＢ121のデータ処理部132はコンピュータＡ101の公開鍵aQの入力を入力メッセージ204として受け付けると、データ処理部132は、記憶部122の秘密情報125から読み出したコンピュータＢ121の秘密鍵bと、コンピュータＡ101の公開鍵aQとをスカラー倍計算部135へ送る(図２の206)。

【０１００】スカラー倍計算部135は、秘密鍵bと公開鍵aQによるスカラー倍点baQを計算し、計算されたスカラー倍点をデータ処理部132へ送る(図２の208)。

【０１０１】データ処理部132は、送られたスカラー倍点を用いて共有情報の導出を行い、記憶部122に秘密情報125として格納する。例えばスカラー倍点baQのx座標を共有情報とする。ここで数abと数baは数値として同じなので点abQと点baQは同じ点となり、同じ情報が導出されたことになる。

【０１０２】ネットワーク142には、点aQと点bQが送信されるが、点abQ(もしくは点baQ)を計算するには秘密鍵aもしくは秘密鍵bを用いなければならない。すなわち秘密鍵aもしくは秘密鍵bを知らなければ、共有情報を得ることができない。このようにして得られた共有情報は、共通鍵暗号の秘密鍵として利用できる。

【０１０３】本実施形態においても、スカラー倍計算部115、135は上述の特徴を備えるので、サイドチャネル攻撃及びフォールト攻撃を防ぐ鍵交換処理が可能になる。

【０１０４】また上記説明における暗号化処理部、復号化処理部、署名作成部、署名検証部および鍵交換処理部は、専用のハードウェアを用いて行ってもよい。またスカラー倍計算部をコプロセッサまたはそれ以外の専用ハードウェアで実現しても良い。

【０１０５】またデータ処理部は、上記暗号化処理、復号化処理、署名作成処理、署名検証処理、鍵交換処理のうち、任意の一つ以上の処理を行えるように構成してもよい。

【０１０６】

【発明の効果】以上述べたように本発明によれば、サイ

ドチャネル攻撃及びフォールト攻撃に対してより安全な楕円曲線演算を用いたメッセージ処理が可能になる

【図面の簡単な説明】

【図１】実施形態におけるシステム構成図である。

【図２】各実施形態における情報の受け渡しを示すシーケンス図である。

【図３】実施形態におけるスカラー倍計算部の構成図である。

【図４】第１実施例のスカラー倍計算方法を示すフローチャート図である。

【図５】第１実施例のスカラー倍計算方法を示すフローチャート図（続き）である。

【図６】第２実施例のスカラー倍計算方法を示すフローチャート図である。

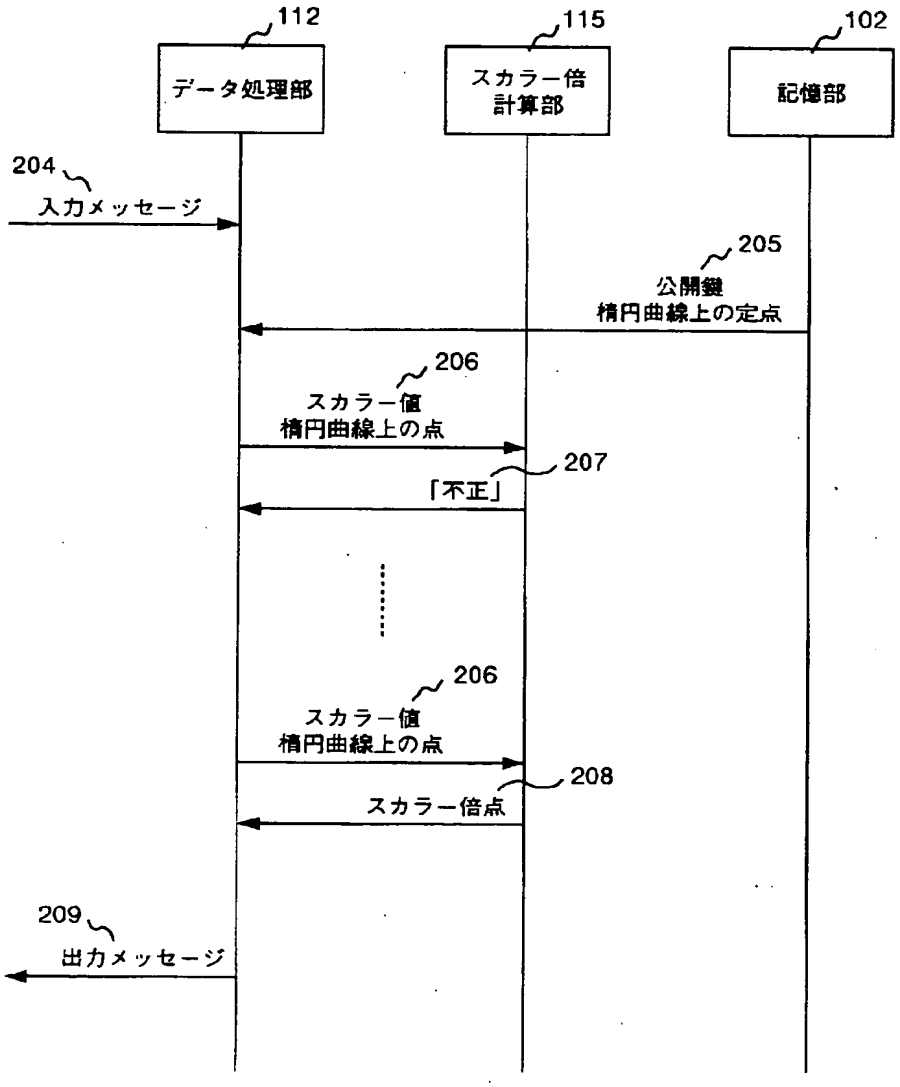
【図７】実施形態における署名検証システムの構成図である。

【符号の説明】

101，121，721：コンピュータ、701：スマートカード、115，135，715，735：スカラー倍計算部、112：データ処理部、132：復号化処理部、712：署名生成処理部、732：署名検証処理部、104，124，704，724：定数、105，125，705，725：秘密情報、301：スカラー倍計算装置

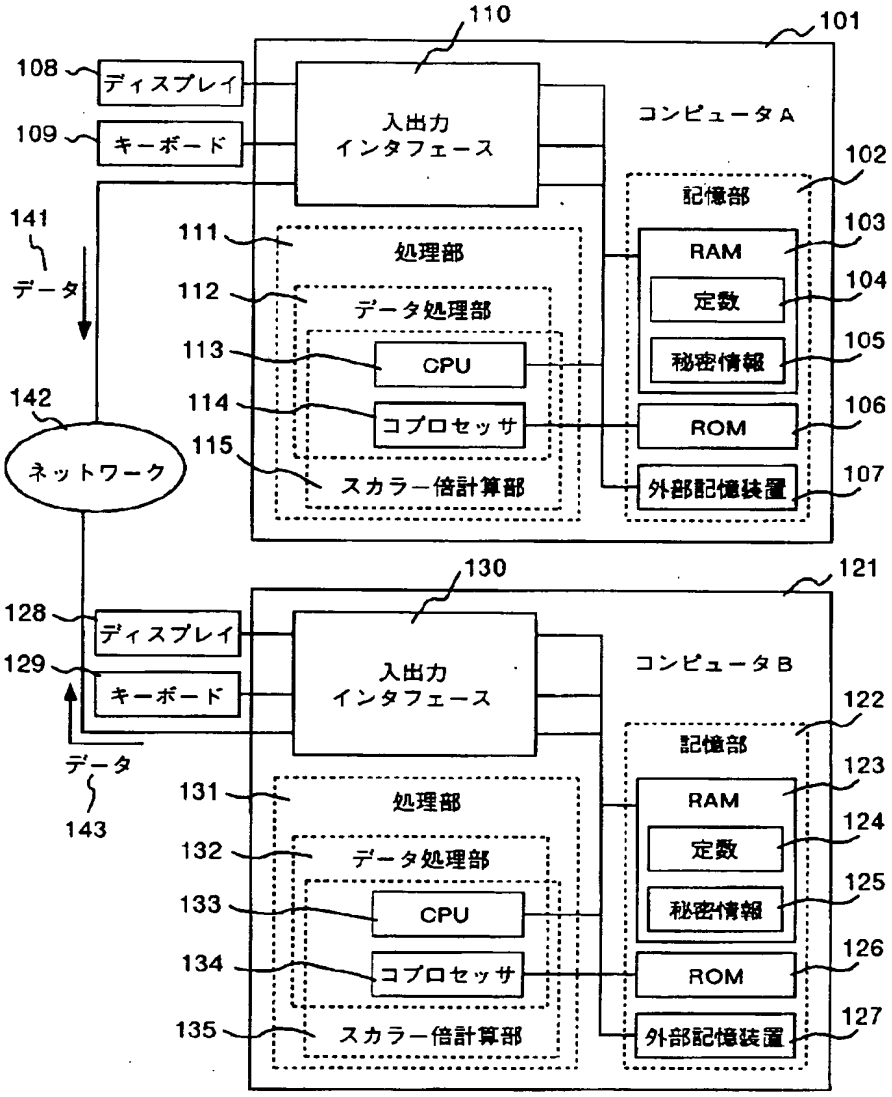
【図２】

図　2



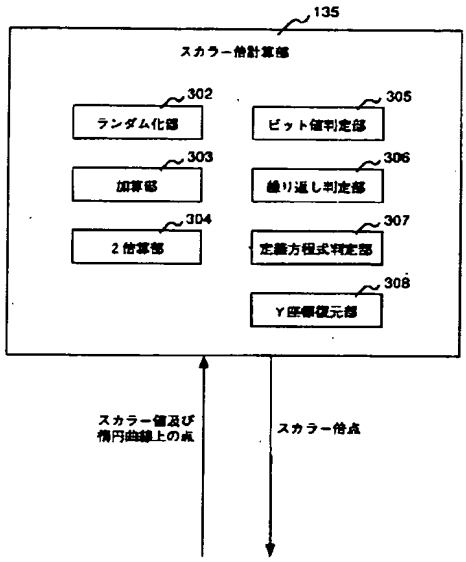
【図１】

図　1


110
101
108
ディスプレイ
109
キーボード
入出力
インタフェース
コンピュータＡ
102
記憶部
141
データ
111
処理部
103
RAM
112
データ処理部
104
定数
113
CPU
105
秘密情報
142
114
コプロセッサ
106
ROM
ネットワーク
115
スカラー倍計算部
外部記憶装置
107
130
121
128
ディスプレイ
129
キーボード
入出力
インタフェース
コンピュータＢ
122
記憶部
データ
143
131
処理部
123
RAM
132
データ処理部
124
定数
133
CPU
125
秘密情報
134
コプロセッサ
126
ROM
135
スカラー倍計算部
外部記憶装置
127

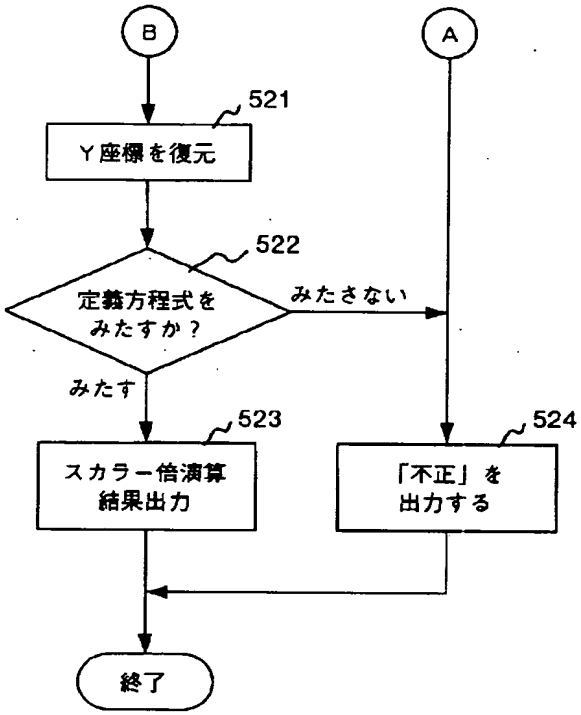
【図３】

図　3


135
スカラー倍計算部
302
ランダム化部
303
加算部
304
2倍算部
305
ビット値判定部
306
繰り返し判定部
307
定義方程式判定部
308
Y座標復元部
スカラー値及び楕円曲線上の点
スカラー倍点

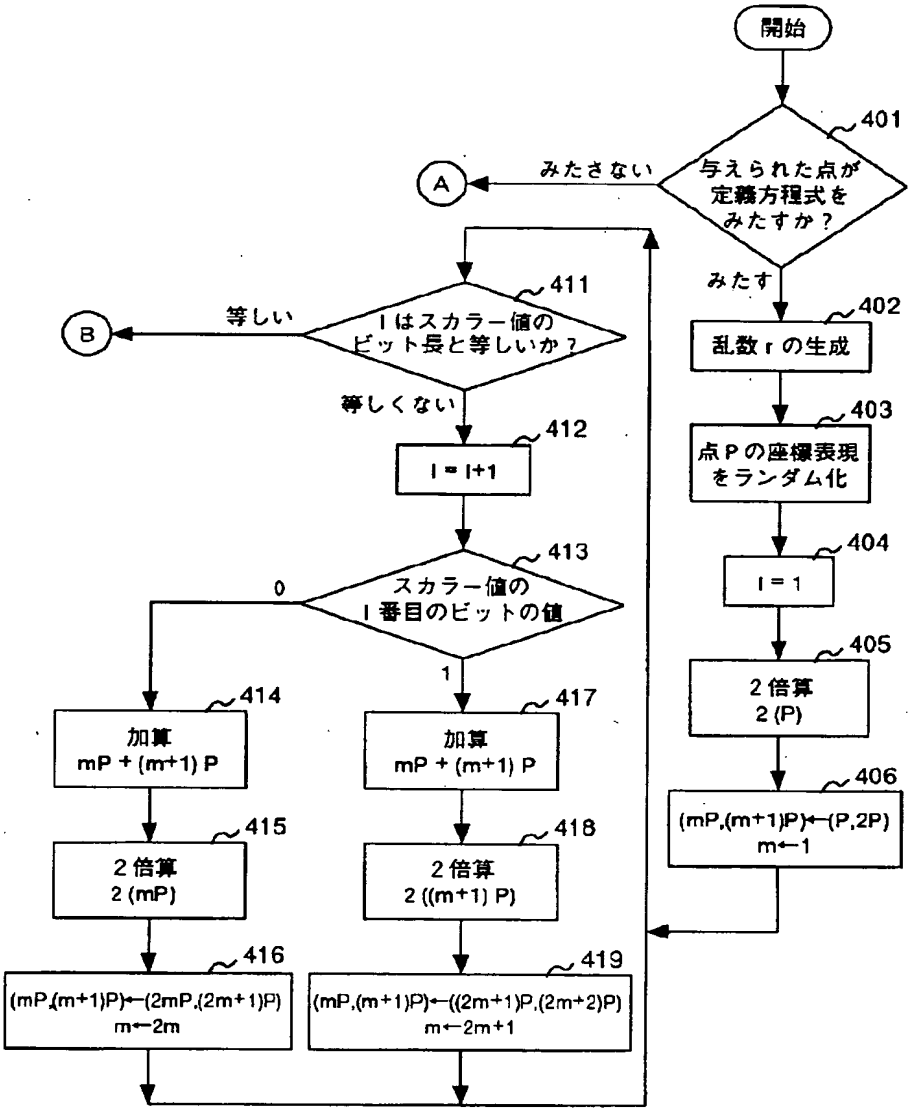
【図５】

図　5


B
A
521
Y座標を復元
522
定義方程式をみたすか？
みたさない
みたす
523
スカラー倍演算結果出力
524
「不正」を出力する
終了

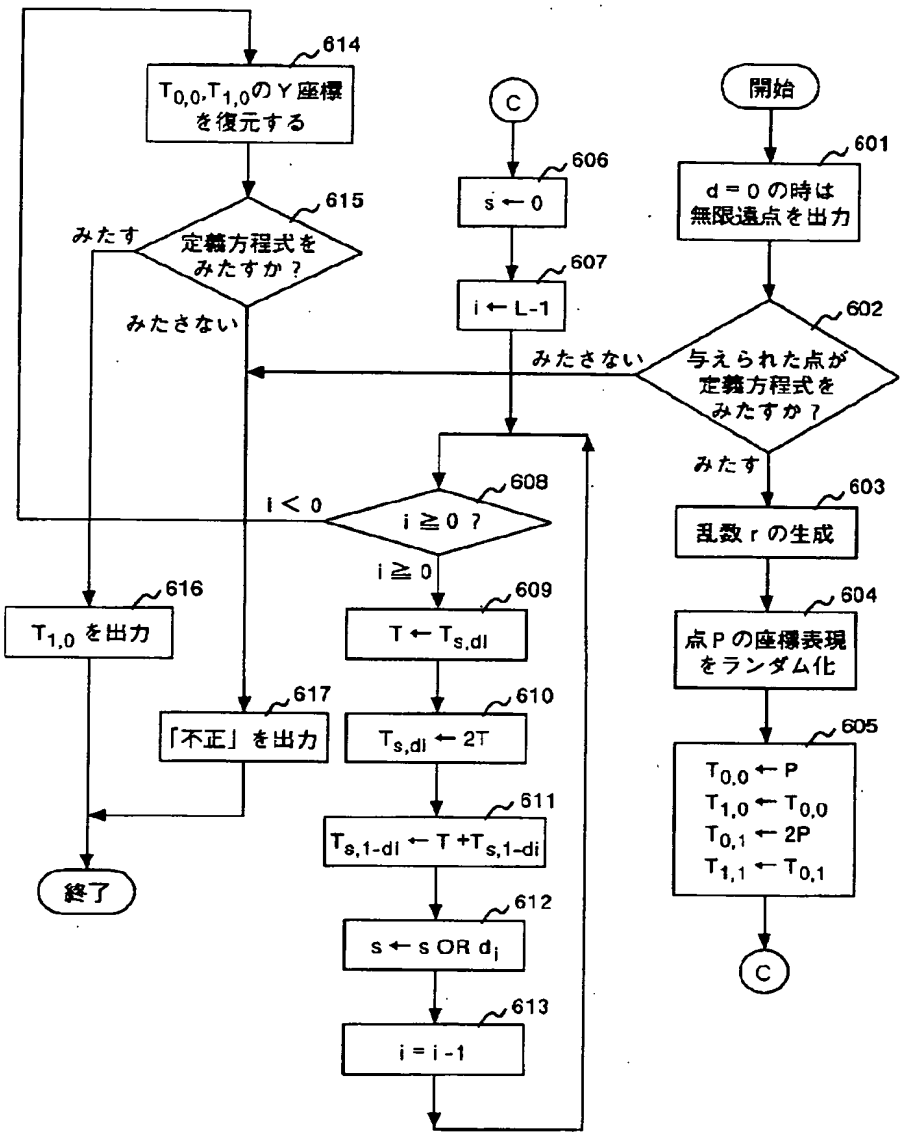
【図４】

図 4


開始
401
与えられた点が定義方程式をみたすか？
みたさない
A
みたす
402
乱数ｒの生成
403
点Ｐの座標表現をランダム化
404
l = 1
405
２倍算 2 (P)
406
(mP,(m+1)P)←(P,2P) m←1
411
lはスカラー値のビット長と等しいか？
等しい
B
等しくない
412
l = l+1
413
スカラー値のｌ番目のビットの値
0
1
414
加算 mP + (m+1) P
415
２倍算 2 (mP)
416
(mP,(m+1)P)←(2mP,(2m+1)P) m←2m
417
加算 mP + (m+1) P
418
２倍算 2 ((m+1) P)
419
(mP,(m+1)P)←((2m+1)P,(2m+2)P) m←2m+1

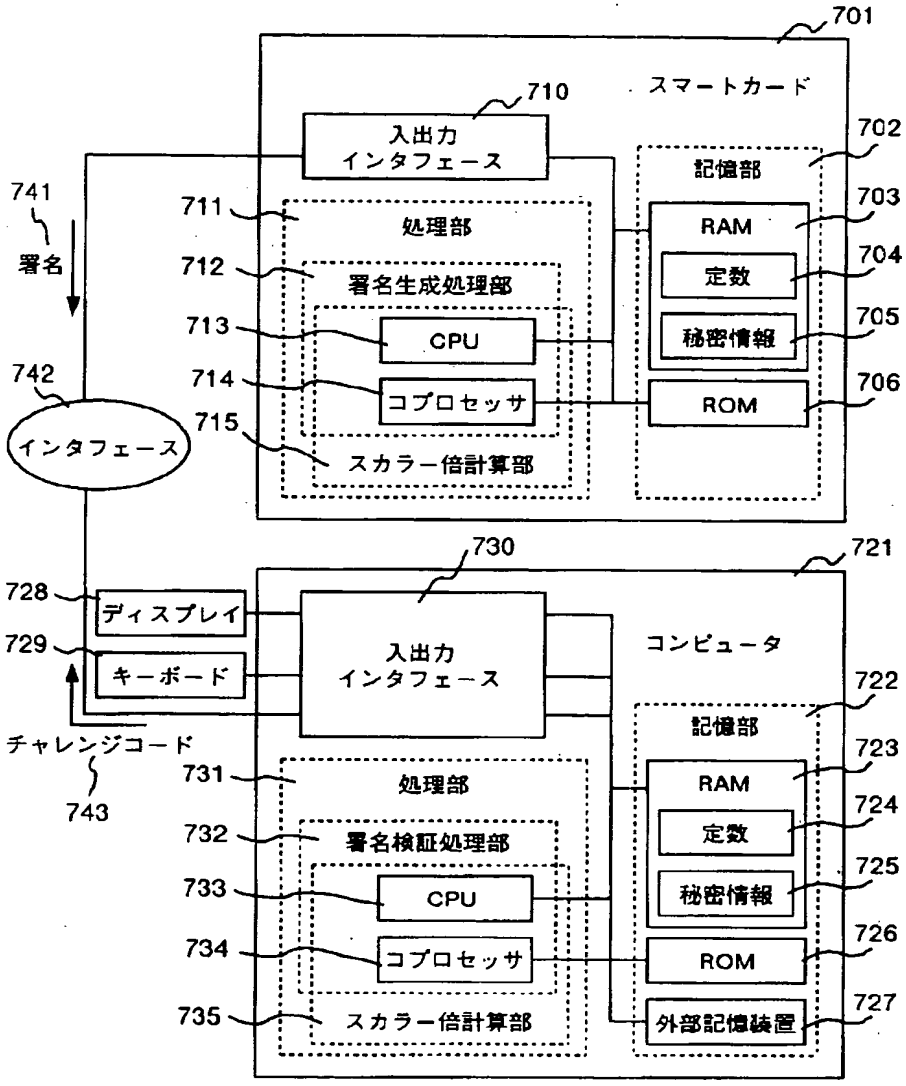
【図６】

図　6


開始
601
d = 0 の時は
無限遠点を出力
602
与えられた点が
定義方程式を
みたすか？
みたす
みたさない
603
乱数 r の生成
604
点 P の座標表現
をランダム化
605
T0,0 ← P
T1,0 ← T0,0
T0,1 ← 2P
T1,1 ← T0,1
C
606
s ← 0
607
i ← L-1
608
i ≧ 0 ?
i < 0
i ≧ 0
609
T ← Ts,dl
610
Ts,dl ← 2T
611
Ts,1-dl ← T +Ts,1-di
612
s ← s OR di
613
i = i -1
614
T0,0,T1,0のＹ座標
を復元する
615
定義方程式を
みたすか？
みたす
みたさない
616
T1,0 を出力
617
「不正」を出力
終了

【図７】

図　7


701
スマートカード
710
入出力
インタフェース
702
記憶部
741
署名
711
処理部
703
RAM
712
署名生成処理部
704
定数
713
CPU
705
秘密情報
742
714
コプロセッサ
706
ROM
715
スカラー倍計算部
インタフェース
730
721
728
ディスプレイ
729
キーボード
入出力
インタフェース
コンピュータ
722
記憶部
チャレンジコード
743
731
処理部
723
RAM
724
定数
732
署名検証処理部
725
秘密情報
733
CPU
734
コプロセッサ
726
ROM
735
スカラー倍計算部
727
外部記憶装置